

1 **DISK DRIVE EMPLOYING MULTIPLE INDEX MARKS**

2
3 **BACKGROUND OF THE INVENTION**

4 **Field of the Invention**

5 The present invention relates to disk drives. In particular, the present invention relates to
6 a disk that employs multiple index marks.

7 **Description of the Prior Art**

8 As shown in FIG. 1, a conventional disk drive employs at least one disk 2 having a
9 plurality of concentric, radially spaced tracks 4. Each track 4 comprises a number of data sectors
10 interrupted by a number of embedded servo sectors 6 used to position a head over a target track
11 and maintain the head over a centerline of the target track during read/write operations.
12 Typically, both sides of the disk 2 are used to record information wherein a respective head is
13 actuated radially over each disk surface. When the disk drive is powered on or a head switch
14 occurs, the embedded servo sectors 6 are located and then used to servo the respective head over
15 the disk 2.

16 Each servo sector 6 comprises a servo address mark (SAM) 8, a preamble 10, a sync mark
17 12, an index mark 14, servo data 16, and servo bursts 18. The SAM 8 comprises a predetermined
18 sequence of bits (e.g., a plurality of "0" bits), which can be detected asynchronously during a
19 power on or head switch event. Once the SAM 8 is detected in one of the servo sectors 6, the
20 location of the remaining servo sectors 6 can be determined and suitable timing circuitry is
21 configured to enable demodulation circuitry at the appropriate time (synchronous with the
22 occurrence of each servo sector). The demodulation circuitry processes the preamble 10 to
23 synchronize timing and gain control circuitry, and the sync mark 12 identifies the beginning of
24 the servo data 16. The servo data 14 comprises information about the track, such as a track ID
25 used during seek operations. Each servo burst 18 following the servo data comprises a "burst" of
26 magnetic transitions, which are typically integrated to generate a servo burst signal. The servo

1 burst signals are evaluated relative to one another to generate a position error signal (PES) used
2 to maintain the head over a centerline of the track (tracking) during read/write operations.

3 Before performing a write operation to a particular track, it is important to accurately
4 locate the target data sector to prevent inadvertently overwriting data in another data sector. To
5 this end, prior art disk drives have employed a disk format wherein each data sector comprises a
6 sector ID used to identify the sector. However, recording a sector ID in each data sector reduces
7 the overall storage capacity due to format inefficiencies. More recent disk drives achieve
8 improvement in format efficiency through an "ID-Less" disk format wherein the location of each
9 data sector is derived from the servo sectors 6. One technique for identifying the data sectors in a
10 track is to consider one of the servo sectors 6 as an INDEX servo sector as shown in FIG. 1. The
11 INDEX servo sector records a valid index mark value (e.g., a "1" bit) in the index field 14, and
12 the remaining servo sectors 6 record a non-index mark value (e.g., a "0" bit) in the index field 14.
13 Once the INDEX servo sector is located, the remaining servo sectors 6 are referenced
14 incrementally (e.g., using a servo sector counter). The data sectors are then identified using
15 suitable timing circuitry after the occurrence of each servo sector 6. However, if the servo
16 system loses synchronization after initially detecting the INDEX servo sector, the error condition
17 is not detected until the INDEX servo sector is detected again. In addition, when the disk drive
18 switches heads in order to access a different disk surface, it may require up to an entire revolution
19 of latency to detect the INDEX servo sector. Still further, if a single bit is used to identify the
20 INDEX servo sector, it may require several additional revolutions to ensure that the INDEX
21 servo sector is not falsely detected due to noise.

22 U.S. Patent No. 5,768,044 discloses a method for reducing the rotational latency after a
23 head switch by adjusting the servo sector counter to account for the skew between the servo
24 sectors on each disk surface. However, this technique also encodes a servo sector number into
25 each servo sector to increase reliability in the event that the servo sector counter is adjusted by
26 the wrong amount. Recording a servo sector number into each servo sector is undesirable
27 because it reduces the format efficiency similar to recording a sector ID in each data sector.

1 There is, therefore, a need to increase servo synchronization robustness as well as reduce
2 the rotational latency during a head switch in a disk drive employing ID-Less data sectors
3 without reducing the format efficiency of the embedded servo sectors.

4 **SUMMARY OF THE INVENTION**

5 The present invention may be regarded as a disk drive comprising a disk surface having a
6 plurality of concentric, radially spaced tracks, wherein each track comprises a plurality of data
7 sectors and a plurality of servo sectors. The plurality of servo sectors comprise a first index
8 servo sector, a second index servo sector, and at least one non-index servo sector between the
9 first and second index servo sectors. A first index mark identifies the first index servo sector and
10 a second index mark identifies the second index servo sector, wherein the first index mark is
11 different than the second index mark. The disk drive further comprises a head actuated over the
12 disk surface, and a disk controller for maintaining a servo sector counter that identifies the
13 circumferential location of the servo sectors. The disk controller detects one of the first and
14 second index marks and initializes the servo sector counter relative to which index mark is
15 detected.

16 In one embodiment, the disk controller detects a loss of synchronization to the servo
17 sectors by detecting one of the first and second index marks at the wrong time, and re-initializes
18 the servo sector counter if loss of synchronization is detected.

19 In another embodiment, the disk drive comprises a first head actuated over a first disk
20 surface and a second head actuated over a second disk surface. The disk controller performs a
21 head switch operation to switch from the first head as the active head to the second head as the
22 active head, and detects one of the first and second index marks recorded on the second disk
23 surface after performing the head switch operation.

24 In yet another embodiment, each servo sector comprises an index mark field for storing a
25 plurality of bits for recording one out of a group consisting of the first index mark, the second
26 index mark, and a non-index mark.

1 In another embodiment, a first plurality of servo sectors comprise information for
2 identifying the first index mark, and a second plurality of servo sectors comprise information for
3 identifying the second index mark. In one embodiment, the first plurality of servo sectors does
4 not include the first index servo sector, and the second plurality of the servo sectors does not
5 include the second index servo sector. In yet another embodiment, each of the first plurality of
6 servo sectors comprise at least one bit of the first index mark, and each of the second plurality of
7 the servo sectors comprise at least one bit of the second index mark.

8 In yet another embodiment, each servo sector comprises a sync mark field for
9 synchronizing to a servo data field, wherein the sync mark field stores one of a first and second
10 sync mark, and the first sync mark is different than the second sync mark. The sync mark field in
11 each of a first plurality of servo sectors identifies one bit of the first index mark, and the sync
12 mark field in each of a second plurality of the servo sectors identifies one bit of the second index
13 mark.

14 In another embodiment, the first and second index marks comprise a sequence of index
15 bits that satisfy a run length limit (RLL) constraint, and a plurality of non-index servo sectors
16 between the first and second index servo sectors comprise a sequence of non-index bits that
17 violate the RLL constraint.

18 In still another embodiment, the first and second index marks are fault tolerant, and in
19 another embodiment the first and second index marks comprise redundancy bits for
20 distinguishing between the first and second index marks.

21 The present invention may also be regarded as a method of operating a disk drive, the
22 disk drive comprising a disk surface having a plurality of concentric, radially spaced tracks.
23 Each track comprises a plurality of data sectors and a plurality of servo sectors, wherein the
24 plurality of servo sectors comprise a first index servo sector, a second index servo sector, and at
25 least one non-index servo sector between the first and second index servo sectors. A first index
26 mark identifies the first index servo sector and a second index mark identifies the second index
27 servo sector, wherein the first index mark is different than the second index mark. A servo sector

1 counter is maintained that identifies the circumferential location of the servo sectors. One of the
2 first and second index marks is detected, and the servo sector counter is initialized relative to
3 which index mark is detected.

4 **BRIEF DESCRIPTION OF THE DRAWINGS**

5 FIG. 1 shows a prior art disk format comprising a plurality of concentric, radially spaced
6 tracks, wherein each track comprises a plurality of data sectors and a plurality of servo sectors,
7 and the servo sectors are referenced circumferentially relative to a single index mark identified by
8 an index servo sector.

9 FIG. 2A shows a disk drive according to an embodiment of the present invention
10 employing a disk format that includes at least a first index mark and a second index mark
11 identified by index servo sectors.

12 FIG. 2B shows an embodiment of the present invention wherein top and bottom heads are
13 actuated by a voice coil motor over top and bottom disk surfaces.

14 FIG. 2C is a flow diagram executed by a disk controller according to an embodiment of
15 the present invention wherein a servo sector counter is initialized relative to which index mark is
16 detected.

17 FIG. 3A shows a disk format according to an embodiment of the present invention
18 including four different index marks distributed around the circumference of the disk.

19 FIG. 3B shows a disk format according to an embodiment of the present invention
20 including eight different index marks distributed around the circumference of the disk.

21 FIG. 4A shows a disk format according to an embodiment of the present invention
22 wherein the bits of four different index marks are distributed over multiple servo sectors.

23 FIG. 4B shows a disk format according to an embodiment of the present invention
24 wherein the bits of a eight different index mark are distributed over multiple servo sectors.

25 FIG. 5 illustrates an embodiment of the present invention wherein each bit of a plurality
26 of index marks is recorded in a respective index servo sector, and the bits of each index mark are

1 recorded according to a run length limit (RLL) constraint to help reduce ambiguity in detecting
2 each index mark.

3 FIG. 6 shows circuitry including a plurality of index detect correlators and a
4 demultiplexer for detecting one of a plurality of index marks and initializing the servo sector
5 counter according to an embodiment of the present invention.

6 **DESCRIPTION OF THE PREFERRED EMBODIMENTS**

7 FIG. 2A shows a disk drive according to embodiment of the present invention comprising
8 a disk surface 20A having a plurality of concentric, radially spaced tracks 4A. Each track 4A
9 comprises a plurality of data sectors and a plurality of servo sectors 6A. The plurality of servo
10 sectors 6A comprise a first index servo sector 6A₁, a second index servo sector 6A₂, and at least
11 one non-index servo sector between the first and second index servo sectors 6A₁ and 6A₂. A first
12 index mark INDEX₁ identifies the first index servo sector 6A₁ and a second index mark INDEX₂
13 identifies the second index servo sector 6A₂, wherein the first index mark INDEX₁ is different
14 than the second index mark INDEX₂. The disk drive further comprises a head 26A actuated over
15 the disk surface 20A, and a disk controller 28 for maintaining a servo sector counter that
16 identifies the circumferential location of the servo sectors 6A. FIG. 2C shows a flow diagram
17 executed by the disk controller 28. At step 32 one of the first and second index marks INDEX₁
18 and INDEX₂ is detected. If the first index mark INDEX₁ is detected at step 34, then at step 36
19 the servo sector counter is initialized relative the first index mark INDEX₁. If the second index
20 mark INDEX₂ is detected at step 34, then at step 38 the servo sector counter is initialized relative
21 the second index mark INDEX₂.

22 In one embodiment, the disk controller 28 detects a loss of synchronization to the servo
23 sectors 6A by detecting one of the first and second index marks INDEX₁ or INDEX₂ at the wrong
24 time, and re-initializes the servo sector counter if loss of synchronization is detected. On average
25 a loss of synchronization will be detected within a quarter a revolution of the disk when two
26 index marks are employed as compared to a half revolution of the disk when only one index
27 mark is employed.

1 In one embodiment of the present invention, the disk drive comprises a single head
2 actuated over a disk surface. In the embodiment of FIG. 2B, the disk drive comprises a first head
3 26A actuated over a first disk surface 20A, and a second head 26B actuated over a second disk
4 surface 20B. The disk controller 28 performs a head switch operation to switch from the first
5 head 26A as the active head to the second head 26B as the active head, and detects one of the
6 first and second index marks recorded on the second disk surface 20B after performing the head
7 switch operation. Employing multiple index marks reduces the rotational latency required to
8 initialize the servo sector counter after a head switch operation. In the example of FIG. 2A, when
9 two index marks are employed the rotational latency is reduced on average to one quarter of a
10 revolution as compared to one half of a revolution when a single index mark is employed. FIG.
11 3A shows an embodiment of the present invention employing four index marks reducing the
12 average rotational latency to 1/8 a revolution, and FIG. 3B shows an embodiment employing
13 eight index marks reducing the average rotational latency to 1/16 a revolution. The number of
14 index marks employed in the embodiments of the present invention is limited by the number of
15 servo sectors 6A, as well as a run length limit (RLL) constraint described in detail below with
16 reference to FIG. 5.

17 The disk drive in FIG. 2A and 2B further comprises a voice coil motor (VCM) 40 which
18 rotates top and bottom actuator arms 42A and 42B about a pivot in order to position the top and
19 bottom heads 26A and 26B radially over the top and bottom disk surfaces 20A and 20B. In one
20 embodiment, the disk controller 28 comprises a read channel for demodulating the read signal
21 generated by the active head into a binary sequence representing the data recorded on the disk
22 surface, as well as a servo controller for generating servo control signals applied to the VCM 40.
23 The read channel and servo controller may be implemented as separate integrated circuits or they
24 may be combined with other disk controller circuitry into a "system on a chip".

25 After performing a head switch operation, the location of the servo sectors recorded on
26 the newly selected disk surface is not synchronized. Therefore the disk controller 28 searches
27 asynchronously for the SAM 8 (FIG. 1) recorded in one of the servo sectors. Once a SAM 8 is

1 detected, the servo sectors can be synchronized so that the disk controller 28 can determine when
2 a next servo sector will pass under the head. However, the location of the data sectors recorded
3 in the track is unknown until one of the index marks is detected. When one of the index marks is
4 detected (INDEX₁ or INDEX₂), the disk controller 28 initializes a servo sector counter relative to
5 which index mark is detected. For example, the disk controller 28 may initialize the servo sector
6 counter to zero if the first index mark INDEX₁ is detected, and to a non-zero value if the second
7 index mark INDEX₂ is detected. After initializing the servo sector counter, as each successive
8 servo sector passes under the head the servo sector counter is incremented, and the location of
9 each data sector within each servo wedge can be determined from the servo sector counter
10 together with other suitable timing circuitry as is well known in the art.

11 In the embodiments of FIG. 2A, 3A and 3B, each servo sector comprises an index mark
12 field 14 (FIG. 1) for storing the bits of the index marks. Each index servo sector comprises a
13 non-zero binary sequence identifying one of the index marks, and each non-index servo sector
14 between the index servo sectors stores a zero binary sequence (no index mark). If two index
15 marks are employed (FIG. 2A), the index mark field 14 comprises two bits for identifying a non-
16 index mark value (00), a first index mark (10), and a second index mark (11). Similarly, three
17 bits are used to identify four index marks (FIG. 3A), and four bits are used to identify eight index
18 marks (FIG. 3B). In an embodiment disclosed below, the index mark field 14 may comprise
19 additional bits to implement fault tolerant detection and/or redundancy bits to help distinguish
20 between the different index marks.

21 In an alternative embodiment, the number of bits in the index mark field 14 is reduced by
22 distributing the bits of an index mark over a plurality of the servo sectors. FIG. 4A shows an
23 embodiment wherein the index mark field 14 comprises a single bit for storing one bit of the
24 index mark. In this example, three bits are used to identify four index marks, wherein three
25 consecutive servo sectors store the three bits of an index mark. The non-index servo sectors
26 store a zero in the index mark field 14. The disk in FIG. 4A rotates counter clockwise such that
27 after detecting two consecutive "0" bits detecting a "1" bit in the index mark field 14 indicates

1 the beginning of an index mark, and the next two servo sectors store the remaining two bits
2 identifying the remainder of the index mark. FIG. 4B shows a similar embodiment wherein four
3 bits are distributed over four consecutive servo sectors to identify eight index marks. This
4 embodiment reduces the number of bits in the index mark field 14 from four to one, which is a
5 significant improvement in format efficiency.

6 In the embodiment of FIG. 4A, the first servo sector in the group that identifies an index
7 mark is considered the index servo sector, and the servo sector counter is initialized relative to
8 this servo sector. However, any of the servo sectors may be considered as the index servo sector,
9 and in one embodiment, a servo sector outside of the group identifying the index mark is
10 considered as the index servo sector for initializing the servo sector counter.

11 In yet another embodiment, each servo sector comprises a sync mark field 12 (FIG. 1) for
12 synchronizing to a servo data field 16, wherein the sync mark field 12 stores one of a first and
13 second sync mark, and the first sync mark is different than the second sync mark. The sync mark
14 field 12 in each of a first plurality of servo sectors identifies one bit of the first index mark, and
15 the sync mark field 12 in each of a second plurality of the servo sectors identifies one bit of a
16 second index mark. For example, the first sync mark may represent a "0" bit, and the second
17 sync mark may represent a "1" bit in the index mark sequence. This embodiment further
18 improves format efficiency by completely eliminating the index mark field 14.

19 In the embodiment of FIG. 4B, the index marks occur one after another making it
20 impossible to determine where each index marks begins. Therefore in one embodiment the index
21 marks comprise a sequence of index bits recorded in consecutive index servo sectors that satisfy
22 a run length limit (RLL) constraint, and a plurality of non-index servo sectors between the index
23 servo sectors comprise a sequence of non-index bits that violate the RLL constraint. This
24 embodiment is illustrated in FIG. 5 which shows the bits of a first index mark INDEX N and a
25 second consecutive index mark INDEX N+1, wherein each servo sector stores one bit of the
26 index mark. The sequence of index bits that identify an index mark satisfy an RLL constraint of
27 three consecutive "0" bits (RLL = 3). The non-index servo sectors between the index servo

1 sectors store a sequence of "0" bits which violates the RLL constraint (i.e., comprises a sequence
2 of "0" bits greater than three). The beginning of an index mark is detected by the first "1" bit
3 that follows a sequence of "0" bits which violates the RLL constraint.

4 FIG. 6 shows circuitry according to an embodiment of the present invention for detecting
5 the different index marks and for initializing the servo sector counter. The sequence of bits
6 representing the read data 46 is input into a number of index detect correlators 48_1-48_N . Each
7 index detect correlator 48 comprises XOR circuitry for correlating the read data 46 with a
8 sequence of bits corresponding to a respective index mark. The output of the XOR circuits is
9 summed, and the summation compared to a threshold. If the summation exceeds the threshold,
10 the index mark is considered detected and a respective output signal 50_1-50_N is activated. A
11 demultiplexer 52 converts the activated output signal 50_1-50_N into a multi-bit control signal 54
12 applied to a load counter circuit 56. The load counter circuit 56 then loads the servo sector
13 counter 58 with an appropriate value corresponding to the index mark detected.

14 In one embodiment, the sequence of bits for identifying the index marks is selected to
15 have a predetermined fault tolerance. This enables a predetermined number of bits within an
16 index mark to be misdetected (due to noise) while still detecting the index mark correctly. In this
17 embodiment, the threshold in the index detect correlators 48_1-48_N is reduced to detect less than a
18 perfect match.

19 In yet another embodiment, the sequence of bits for identifying the index marks includes
20 redundancy bits for implementing an error correction code (ECC), such as a Hamming code or
21 other suitable ECC code. This allows bits that are misdetected due to noise to be corrected,
22 thereby helping to distinguish between the different index marks.